CIS 730 Artificial Intelligence
CIS 530 Principles of Artificial Intelligence
Fall 2007

Homework 1 of 10: Problem Set (PS1)
Warm-up: Intelligent Agents, Search, Game Tree Search

Assigned: Wed 22 Aug 2007
Due: Wed 05 Sep 2007 (before midnight)

The purpose of this assignment is to exercise your basic understanding of intelligent agents, state space search, and game theory, and to help you apply these concepts simulate the behavior of search algorithms.

This homework assignment is worth a total of 20 points.
Each problem is worth 2 points for CIS 730 students and 3 points for CIS 530 students.
Turn in hard copy or attach an electronic copy of the assignment in PDF form (converted from your word processor, or scanned) to the instructor at: **CIS730TA-L@listserv.ksu.edu**

1. **(530/730) Bounded rationality.** Continuing the class discussion from Wed 22 Aug 2007: what limitations of reasoning, specifically the ability to access knowledge about the state of the world and the (probable) effects of actions, differentiate a boundedly rational agent from an omniscient one? Give a concrete example of such a limitation.

2. **(530/730) State space representation.** (Adapted from Problem 3.9a, p. 90 R&N 2e and Winston 3e.) The Farmer, Fox, Goose and Grain (FFGG) problem is usually stated as follows:

   > *A farmer comes to a river bank with a fox, a goose, and a sack of grain. He can cross over using a boat that holds himself and one of the three items at a time. If the fox is left on one bank with the goose, it will eat the goose. If the goose is left on one bank with the grain, it will eat the grain. Can the farmer get all three items and himself across without anything being eaten?*

   Formulate the problem precisely as a state space. Turn in an illustration of the state space diagram, showing which states are the initial states and which are reachable from each other.

3. **(730 only) State space search.** Write a simple program in any programming language to implement the data type for the above problem and for committing moves. Turn in your source code and an interactive test execution in which **you** solve the problem by moving the right items across with the farmer. You need not actually implement a solver, but write a short paragraph describing how you would do it.

4. **(530/730) Beam search.**

   a. What type of beam search specification is equivalent to best-first search? Prove your answer.
   b. Give an example of a graph for which hill-climbing and beam search with a beam width of $w = 2$ produce different answers.

5. **(530/730) Constraint Satisfaction.** (From http://snipurl.com/1pu4q; see also http://snipurl.com/vfl2). Solve the following cryptarithmetic problem by hand, using backtracking and forward checking.

   Find an assignment of the integers 0-9 to the letters in the words SEND, MORE, and MONEY such that:

   a. each integer is assigned to a unique letter
   b. each letter is assigned to a unique integer
   c. the assignment satisfies the equation ***SEND + MORE = MONEY***
   d. 1 is assigned to the letter M

   Show your work *in the manual solution.*

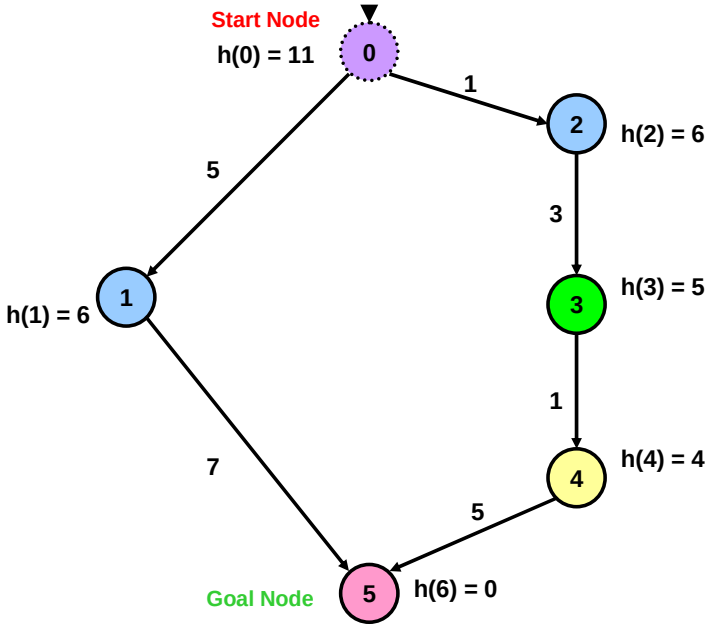
6. **(730 only) Heuristic search (adapted from Problem 4.11, p. 135 R&N 2e) Give the name of the algorithm that results from each of the following special cases:**

   a. Simulated annealing with T = 0 at all times (and omitting the termination test).
   b. Genetic algorithm with population size N = 1.

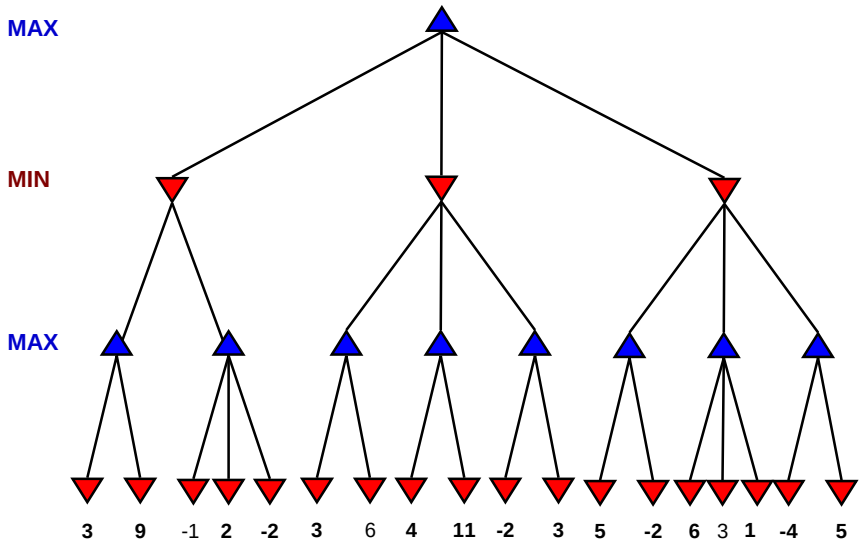
7. **(530/730) PEAS representations of problems.** For each of the following software agent tasks, develop a Performance measure, Environment, Actuators, Sensors (PEAS) description of the task environment:

   a. bidding for a specified (but not necessarily unique) item through an online *second-price auction* system such as eBay – see http://en.wikipedia.org/wiki/Online_auctions
   b. increasing the level of a character in a massively multiplayer online role-playing game (MMORPG) such as *World of Warcraft* – see http://en.wikipedia.org/wiki/MMORPG

8. **(530/730) Heuristic Search.** Simulate the behavior of A* on the following graph, showing the **nodes expanded,** the **path** actually returned, and the **cost** of the path.



9. **(530/730) Game Tree Search.** Solve the following tree using minimax and alpha-beta pruning. Mark what states are pruned (not evaluated) in the latter case.



10. **(730 only) Games.** For the Angband computer game we looked at in class, why might using a game tree (or expectiminimax tree) not be practical? What can be used instead?

Class participation (required).

Consider a coal-mining robot. Besides cost and payoff, what are measures of utility that could be incorporated into such an agent, and in what scenarios could they apply?

Post your discussion to **CIS730-L@listserv.ksu.edu**, along with a brief introduction stating your:

- name
- program (grad or undergrad) and major
- interests in AI
- special topics you would like to see covered, if any